International Journal of Innovation Scientific Research and Review
Vol. 07, Issue, 04, pp.8129-8138, April 2025
Available online at http://www.journalijisr.com
SJIF Impact Factor 6.599

Research Article

ISSN: 2582-6131

OPTIMIZING IOT DATA MANAGEMENT WITH ADVANCED DATABASES

[1,*] Ayad Zedo Ismaeel and [2]Ibrahim Mahmood Ibrahim

[1]Akre University for Applied Sciences, Technical College of Informatics, Department of Information Technology, Duhok, Iraq.
[2]Computer Networks and Information Security.

Received 11th February 2025; Accepted 12th March 2025; Published online 20th April 2025

ABSTRACT

The expanding integration of the Internet of Things (IoT) into numerous industries has boosted need for efficient, scalable, and secure data management solutions. Traditional database approaches are frequently unable to handle the dynamic, high-volume, and diverse data produced by IoT devices. This analysis examines the important breakthroughs in database technology designed for IoT contexts, with an emphasis on architectural improvements, query processing methodologies, performance optimization tactics, and security upgrades. The research emphasizes the value of real-time data processing, low-latency access, and flexibility to both centralized and decentralized infrastructures. Furthermore, the research highlights the need of cloud-edge integration and intelligent resource management in maintaining responsiveness and efficiency. This article gives a complete overview of current trends as well as insights into future prospects for developing strong data infrastructures in IoT ecosystems by assessing various methodologies and system capabilities.

Keywords: Internet of Things (IoT), Database Management Systems (DBMS), NoSQL Databases, Real-Time Data Processing, Edge and Cloud Computing, Data Security and Privacy, IoT Data Optimization.

INTRODUCTION

IoT devices create massive volumes of data, frequently in real time, requiring efficient storage, retrieval, and analysis. Traditional relational databases (RDBMS) fail to provide the necessary scalability and flexibility for IoT applications. NoSQL databases, with their schema-less architecture, have emerged as a viable alternative, providing increased scalability and handling of diverse data.[1][2] The necessity of modifying database management systems (DBMS) to manage the increasing amounts of data produced by Internet of Things (IoT) systems is examined in the introduction. In order to solve issues like security, concurrent data access, and query performance optimization—all of which are critical for applications that need instant data processing—it places a strong emphasis on the development of real-time DBMS. In order to lessen dependency on real-time DBMS, the study investigates whether correctly designed conventional DBMS can handle IoT data efficiently. The authors focused on improving data accessibility for machine learning, statistical analysis, and Knowledge Discovery in Databases (KDD) by simulating concurrent data writing to assess performance. The goal of the project is to increase data availability and enhance concurrent access for IoT-based data mining applications.[3].The difficulties of handling the enormous volumes of data produced by IoT devices in fog and cloud computing settings are discussed in the work. The "Foggy Weather Architecture with Tangential Data Control System" is a revolutionary system that use fog nodes to filter and monitor IoT data in order to address these problems. The system's dynamic algorithm determines whether to process data in the cloud or the fog, giving priority to fog computing during steady times and cloud processing during moments of fast data changes. This method advances IoT applications in cloud settings by optimizing resource utilization, enhancing responsiveness, and increasing the scalability and efficiency of IoT data management.[4] The introduction highlights issues including security, scalability, and transparency in handling IoT data while discussing the substantial influence of the Internet of Things (IoT) on data collecting, processing, and storage. The authors suggest an optimized blockchain architecture that use sharding and pruning strategies to enhance scalability and handle massive data volumes from several IoT devices in order to address these problems. Although the concept has promise, the introduction points out that further study is required to confirm how well it works in practical situations and to fortify its privacy and security features. This lays the groundwork for creating safe and effective IoT data management systems.[5]

The introduction addresses the problems and significance of data management and analytics inside the Internet of Things (IoT). It characterizes IoT as a network of interconnected devices that produce extensive data and emphasizes the necessity for efficient data management, encompassing data collection, storage, processing, and analysis. Principal challenges encompass scalability, data integrity, and interoperability. Analytical methodologies such as machine learning and predictive analytics facilitate the extraction of useful insights, empowering firms to make data-driven choices. Addressing challenges associated with interoperability, privacy, and data quality is essential for utilizing IoT data to foster innovation and change across sectors.[6] The significance of backend databases in facilitating Internet of Things applications—which produce enormous volumes of data—is emphasized in the introduction. It emphasizes the necessity of dependable, safe, and successful IoT systems, emphasizing that database performance is essential to the success of IoT services. In contrast to standard database applications, the research addresses particular issues by measuring database performance in IoT contexts. Databases must increasingly satisfy increased performance standards as technology develops, necessitating constant study and improvement.[7] The opening stresses how important backend systems are for handling the huge amounts of data that IoT devices produce. It shows how important it is to have IoT apps that work well, are safe, and are stable. It also shows how important database speed is for making sure IoT services work well. The study's goal is to compare how well different databases work in IoT settings, taking into account the special problems these settings have compared to regular database

***Corresponding Author: Ayad Zedo Ismaeel,**
1Akre University for Applied Sciences, Technical College of Informatics, Department of Information Technology, Duhok, Iraq.

programs. As technology improves, more IoT apps will need databases that work faster. This shows how important it is to keep researching and coming up with new ideas.[8] The opening talks about how AI, Machine Learning (ML), the Internet of Things (IoT), Blockchain, and Big Data analytics have changed Business Intelligence (BI). BI systems used to only do simple analysis, but now they can also provide real-time, predictive, and automated intelligence to help businesses stay ahead of the competition. These technologies improve BI by offering tracking in real time, prediction analytics, and safe, clear data management. But adding them to BI tools is hard because of issues like data safety, scalability, and interoperability. The study's goals are to look into how these technologies can be used to make BI better, find problems with integration, come up with a plan for successful convergence, and look at examples of AI-driven BI projects that have worked well.[9] The opening talks about how hard it is to handle the huge amounts of time series data that are being created by the growing IoT environment. A lot of study has been done on how to reduce integer value data, but not much on how to handle and retrieve floating-point data. To fix these problems, the study suggests a way to index floating-point time-series data within a compression method that doesn't lose any information. This method makes it easy to get data using timestamps without having to fully extract it, which makes storage and access more efficient. The opening also talks about plans for experiments to show how better storage efficiency and real-time data access can be achieved.[10] The introduction emphasizes the necessity of contrasting MongoDB and InfluxDB's scalability and performance in order to manage time-series data in Internet of Things contexts, where high-concurrency workloads are typical. Although both databases are widely used, little study has been done on how well they work in these kinds of situations. Using a Python-based client script, the study seeks to assess latency, resource consumption, and scalability in order to provide insightful information that will help choose the best database for IoT data management.[11]The introduction emphasizes the Internet of Things' expanding influence and the difficulties in handling the massive amounts of data produced by IoT devices. Due to latency problems with traditional cloud-based data management, edge and fog computing are being adopted for quicker, localized data processing. The study suggests a novel data interaction model to improve performance efficiency across the IoT data life cycle and highlights the significance of a cooperative data flow between edge and cloud to optimize data handling.[12] The introduction emphasizes the increasing expansion of IoT devices and the necessity for effective data processing to handle their massive data volumes. The article suggests edge computing for quick, local data processing to improve IoT network performance. Edge node selection, data preparation, distributed analytics, and dynamic resource allocation are part of the authors' security and privacy-focused strategy. They also suggest further study and evaluate their strategy to enhance IoT networks and encourage innovation.[13]The fast growth of IoT systems presents data management issues. It stresses the importance of efficient DBMS for real-time data processing, security, and concurrent access. The study examines whether correctly designed common DBMS can enable IoT edge devices by improving data retrieval, managing concurrent access, and boosting user and data mining data availability. The project intends to increase real-time data processing and IoT application efficiency.[14] IoT devices like sensors and mobile devices generate continuous data, which presents issues. Due to heterogeneous and high-velocity data, effective data processing and integration are needed. Due to the complexity of IoT data, conventional frameworks typically struggle with query replies, therefore an effective indexing system is crucial. Analyzing streaming data situations and optimizing indexing strategies, the research develops a system to handle these difficulties. Experiments will demonstrate the solution's efficacy compared to current approaches.[15] The introduction emphasizes the importance of machine learning (ML) in interpreting IoT data. By recognizing abnormalities and malicious assaults, ML increases real-time data processing, decision-making, and security. Big data analytics and ML improve IoT data analysis. The study also recommends researching sophisticated ML techniques and data preprocessing to enhance IoT data analysis using real-time sensors like DHT11.[16] IoT services are improving urban life and operational efficiency in smart cities, according to the introduction. It emphasizes the necessity for dataset optimization to improve smart city decision-making prediction processes. The study addresses a gap in IoT and smart city research on optimization methods. PSO can filter and optimize datasets before training, boosting prediction accuracy and efficiency. Their technique will be validated through simulations to deliver accurate and timely smart city IoT solutions.[17] The introduction discusses how standard cloud computing cannot manage IoT device data, causing latency and capacity concerns. Smart city applications require real-time data processing, and edge computing reduces latency and optimizes bandwidth by processing data closer to the source. Edge nodes with machine learning improve real-time analysis and decision-making. The report also addresses edge computing security and resource distribution issues and suggests further research in edge AI and renewable energy for sustainable IoT solutions [18].

BACKGROUND THEORY

1. Internet of Things (IoT) and Data Management

The Internet of Things (IoT) is a network of interconnected physical objects that gather and share data using integrated sensors, software, and communication technology. As IoT expands into industries such as healthcare, manufacturing, smart cities, and transportation, handling the massive and diverse data created becomes important. Real-time data management is critical for ensuring that IoT applications make timely decisions, respond quickly, and operate optimally. The key issues are the diversity of data formats, resource limits, and the necessity for efficient storage and processing procedures.[19][20][21]

2. Machine Learning and AI for Data Optimization

Artificial intelligence (AI) and machine learning (ML) have transformed data management tactics for IoT devices. Techniques like anomaly detection, predictive analytics, and classification models help to improve real-time decision-making and system efficiency. Frameworks like the Machine Learning Analytic-Based Data Classification Framework (MLADCF) use hybrid models to optimize energy usage and resource allocation in restricted contexts. AI also dramatically improves transactional integrity and data security in distributed database systems by anticipating conflicts and allowing self-optimization techniques.[22][20]

3. Big Data and Business Intelligence Integration

Big Data has emerged as a fundamental component of Business Intelligence (BI) frameworks as a result of the proliferation of Internet of Things devices. An intelligent and automated decision-making process may be facilitated by integrating Big Data with artificial intelligence, machine learning, and the internet of things. Blockchain provides additional assistance for the integrity and security of data. Collectively, these technologies make it possible for current business intelligence systems to go beyond traditional descriptive reporting to proactive, self-sufficient decision support systems.[23]

4. Distributed and NoSQL Databases in IoT

Due to scalability and flexibility restrictions, traditional relational databases frequently fail to manage IoT data effectively. NoSQL databases, such as MongoDB, Cassandra, and Redis, provide high availability, horizontal scalability, and are better suited to unstructured and heterogeneous data. Performance tests show that MongoDB frequently outperforms relational systems such as MySQL in terms of latency and throughput, particularly in cloud contexts. Docker-based solutions of distributed databases have demonstrated promising results for Industrial IoT, providing containerization benefits for deployment and scalability.[24][25][26]

5. Real-Time Processing and Streaming Data

In IoT networks, the capacity to interpret data in real time is essential. The TSBPS model and other streaming architectures and frameworks that use spatial-temporal chunking improve data input, storage, and retrieval performance. Real-time systems require not only speed, but also the ability to handle volatile data through effective caching and indexing algorithms. These solutions eliminate system latency and facilitate applications like autonomous driving and intelligent surveillance, which rely on instant input from sensor data.[27]

6. Privacy and Security in IoT Data

The Internet of Things (IoT) presents substantial difficulties to privacy and security due to its widespread nature. It is possible for the personal information of users to be exposed by continuous surveillance and monitoring. In accordance with legislation such as the General Data Protection Regulation (GDPR), privacy-preserving identifiers such as pseudonyms are utilized in order to reduce these dangers. Identity management systems (IdM) that allow for regulated re-identification and incorporate pseudonymity are one of the most important factors in ensuring that data minimization and accountability are maintained. In addition, the safety of sensitive Internet of Things data is ensured by security mechanisms such as encryption, role-based access, and secure transmission protocols.[28]

LITERATURE REVIEW

Zhou et al. (2024) investigated the incorporation of artificial intelligence with edge computing in Internet of Things (IoT) systems, with a particular focus on the ways in which AI-driven decision-making might improve the efficient processing of real-time data. As a result of their research, they discovered that while edge computing helps to increase security and reduce latency in Internet of Things applications, integrating it with artificial intelligence further improves scalability, performance, and autonomy. In addition, they identified issues such as restricted computer resources and decreased energy efficiency, both of which need to be addressed through the utilization of hardware accelerators and federated learning strategies.

Dias et al. (2019) an investigation on the efficiency of NoSQL databases, in particular Cassandra, in managing time-series data from the Internet of Things was carried out. Their findings indicated that the right adjustment of database compaction algorithms may have a considerable influence on both the response time and the storage efficiency of operations. Their findings highlight the significance of matching database configurations with properties of Internet of Things data, such as entries that are organized, time-stamped, and seldom updated, in order to guarantee effective storage and retrieval in large-scale systems.

Kang and Song (2024) We introduced a system that makes use of multi-versioned data semantics and LSM-tree structures in order to optimize time-series searches in Internet of Things databases. In order to enhance query performance and limit the number of unwanted data merges, they used algebraic approaches. This was especially important when dealing with delayed, duplicated, and repaired data. According to the findings of their research, employing version-aware operations inside query plans has the potential to greatly improve operational efficiency in industrial Internet of Things environments.

Wang et al. (2019) It was suggested that actor-oriented databases (AODB) be utilized for the modeling of Internet of Things data systems. Their strategy intended to manage data from numerous Internet of Things devices that were very dynamic and diverse by utilizing communication that was dispersed and asynchronous. Through the use of case studies, the authors proved that AODBs provide a flexible and scalable architecture that is ideal for a wide variety of Internet of Things applications, particularly those that require real-time interactions and decentralized centralized control.

Gadde (2022) We investigated the function that artificial intelligence plays in the process of dynamic data sharding for huge databases. Better load balancing, lower latency, and higher throughput are the results of his introduction of a methodology that is based on machine learning and dynamically updates data partitions depending on real-time workloads. According to the findings of the study, artificial intelligence has the ability to not only improve database performance but also cut down on the amount of operational overhead required to manage big, dispersed data systems.

Lo et al. (2019) a comprehensive literature assessment on the application of blockchain technology in Internet of Things contexts was carried out. Their investigation classified the current solutions into two categories: those that addressed the difficulties of data management and those that addressed device (thing) management. They came to the conclusion that although blockchain technology has a number of benefits, including decentralization and immutability, its incorporation into the Internet of Things (IoT) is still confronted with challenges concerning performance, scalability, and implementation maturity.

Yuan (2024) The problems and techniques for incorporating artificial intelligence into Internet of Things systems were reviewed, with a particular emphasis on real-time decision-making and contexts with limited resources. For the purpose of enabling effective deployment of artificial intelligence on low-power Internet of Things devices, the article examined several strategies such as model compression, quantization, and edge computing. It was stressed that lowering latency and energy consumption is essential for the success of AI-driven Internet of Things systems in applications such as emergency response and predictive maintenance.

Judvaitis et al. (2024) a data management system that places an emphasis on privacy, scalability, and configurability was presented for the Internet of Things (IoT)–Edge–Cloud progression. The fact that their approach incorporates visualization capabilities in addition to differential privacy and energy-efficiency features makes it suited for monitoring critical infrastructure. In order to guarantee deployments that are both secure and effective, the research demonstrates the advantages of integrating data flow management across different types of computing systems at the same time.

Zahedinia et al. (2023) IoT data caching techniques were explored in Named Data Networking (NDN) for the purpose of improvement. In order to address the ephemeral nature of Internet of Things (IoT) data, they devised a caching mechanism that was based on the data lifespan and the position of the node. In order to improve the effectiveness of caching in Internet of Things settings, their technique attempted to minimize redundant operations and maximize memory use.

Nambiar and Mundra (2022) examined the similarities and differences between data lakes and data warehouses as core technologies for business data management. They made the observation that data lakes provide flexibility for storing raw data types that are different, but data warehouses are appropriate for organized analytics that are driven by a specific objective. Through their work, they brought to light the significance of aligning storage strategies with the data requirements of an organization, particularly in the context of big data analytics and the integration of the internet of things.

Lee et al. (2023) in Internet of Things contexts, a data access control and key agreement system that is based on blockchain technology was presented. Their method combines blockchain technology with Ciphertext-Policy Attribute-Based Encryption (CP-ABE) in order to guarantee granular access control and the protection of users' privacy. Additionally, the system offers techniques for mutual authentication and key agreement, which guard against a variety of security risks such as tracking and guessing attacks. Additionally, it enables efficient and reliable data outsourcing and access.

Cooper and James (2009) emphasized the significant difficulties that are associated with the maintenance of databases for the Internet of Things (IoT). They stressed how difficult it is to manage many sorts of data, such as environmental, identity, and location data, which are created by a large number of devices that are connected to one another. The most important topics that were highlighted were data indexing, querying, transaction processing, and integration across a variety of platforms. These are all critical for allowing Internet of Things settings that are both efficient and scalable.

Poornima et al. (2023) data management for Internet of Things (IoT) and Digital Twin (DT) settings was investigated, with a particular emphasis placed on the incorporation of real-time sensing, autonomous control, and analytics. According to what they observed, Digital Twins are able to synchronize with entities that exist in the actual world in order to facilitate decision-making and optimization in smart manufacturing. For the purpose of supporting modeling, simulation, and real-time operations, the study highlighted the difficulties that are caused by latency, hardware limits, and fragmented data, and it proposed solutions that are data-centric and driven by artificial intelligence.

Hewa et al. (2022) suggested a security architecture for cloud manufacturing and 5G-enabled Industrial Internet of Things (IIoT) computing that is based on fog computing and blockchain technology. By moving security functions to the fog layer, their hybrid approach removes the possibility of a single point of failure and decreases the amount of delay experienced. In order to provide trust, data privacy, and low-latency secure communications between IIoT nodes and cloud services, the system incorporates capabilities such as dynamic certificate creation and symmetric key agreements.

Al-Atawi (2024) established a hybrid architecture known as Interconnected Intelligence (II), which combines Internet of Things (IoT), cloud computing, and fog computing. The goal of this architecture is to improve data management and decision-making in real time. By utilizing this design, latency is decreased, resource efficiency is improved, and data security is strengthened. In real-world testing, performance gains were exhibited in decision accuracy, energy economy, and system scalability. These enhancements validated the usefulness of the system across both urban and industrial Internet of Things installations.

Andronie et al. (2023)A comprehensive evaluation of the Internet of Robotic Things (IoRT) was carried out, during which the use of large data management methods, deep learning-based object recognition, and geographic simulation tools were analyzed. Within the context of Internet of Things settings, they placed particular emphasis on the role that machine learning and sensor fusion play in improving autonomous decision-making and real-time monitoring capacity. According to the findings of their research, simulation-based digital twins and federated learning have the potential to be effective in the coordination of complex robotic systems.

Clavijo-López et al. (2024)an Intelligent Database Management System (ML-IDMS) that is based on machine learning was presented for use in scenarios including big data analytics and the internet of things. For the purpose of providing real-time data retrieval, intelligent decision-making, and optimal resource use, the system incorporates machine learning techniques into database management system designs. The findings demonstrated enhanced metrics in query execution, data correctness, and network performance, which exemplifies the potential of machine learning-based intelligent data management systems to reimagine existing intelligent data systems.

Doan et al. (2020) proposed a system that combines lossless compression with efficient indexing in order to handle the issues that are associated with handling streaming data from the Internet of Things. Their technique enables the integration of data in real time from a variety of sources and permits the retrieval of information without going through the process of full decompression. Through the utilization of timestamp-based indexing inside compressed datasets, their solution dramatically improves both the efficiency of storage and the throughput of queries being executed.

The fast proliferation of the Internet of Things (IoT) has resulted in an increase in data production, necessitating resilient and scalable database systems adept at managing substantial, real-time, and diverse data. Conventional relational databases frequently fail to satisfy these requirements, leading to the implementation of more adaptable, distributed, and sophisticated data management solutions. This table provides a comparative examination of several database structures in the context of IoT. It emphasizes essential aspects such query language, optimization techniques, security protocols, performance indicators, and compatibility with cloud and edge installations. This comparative analysis seeks to elucidate the advantages, drawbacks, and appropriateness of each system for diverse IoT contexts, providing insights into the utilization of sophisticated databases to enhance data administration in progressively intricate and data-heavy settings.

Table 1: "Database Systems and Architectures for Internet of Things Data Management: A Comparative Analysis"

#Ref	Database Type & Name	Query Language	Optimization Techniques	Security Mechanisms	Performance Metrics	Cloud/Edge Support
[29]	"MongoDB (Time Series Database)" cell	"MongoDB Query Language (MQL)" cell	Streaming ETL, Filtering Mechanism (value alteration, precision filter), Redis Streams for analytical purposes, Kubernetes for orchestration. Terraform for infrastructure administration	Data localization (edge processing), diminished transmission for privacy, facilitation of federated learning, and cognizance of security vulnerabilities in open contexts.	Round Trip Time (RTT), Median, and Standard Deviation under varying latency and device circumstances	Affirmative; facilitates hybrid edge-cloud architecture, Kubernetes and Proxmox on Edge, containerized deployment, as well as integration of AI and federated learning.
[30]	NoSQL (Cassandra)	Cassandra Query Language (CQL)	TWCS, DTCS, compaction window size tweaking, column family modeling using TTL and static columns	Fundamental deletion using tombstones, with no particular advanced security addressed.	Latency (read/write), Throughput (ops/sec), Disk Space Usage	Not defined; emphasis is on the performance of distributed NoSQL databases, excluding cloud integration.
[31]	Time-Series Database (Apache IoTDB)	SQL-like (Extended with version-aware operators)	VTSA: version reduction, operator pushdown, branch merging, selective updates, relational reducibility.	Not explicitly mentioned; emphasis on query optimization and version control	Latency, disk I/O, CPU use, and throughput (for various query types).	Partially covers IoT deployment possibilities in edge devices and gateways, but not direct cloud interaction.
[32]	Actor-Oriented Database (AODB using Orleans Runtime)	Declarative querying is constrained; access is facilitated using asynchronous actor techniques.	Virtual Actors, In-memory processing, Favor local placement, Actors aggregate, Balancing actor granularity Workflow-based consistency	Actor authentication and access control; state encapsulation and modular actor logic for protection	Latency (including percentiles),Through put (requests per second), Scalability (sensor quantity and server expansion), CPU utilization	Indeed; cloud-based deployment with Amazon AWS facilitates scale-out through server silos, specifically engineered for SaaS applications.
[33]	Distributed NoSQL Database (Cloud-native)	NoSQL queries; not expressly identified, maybe proprietary or standard NoSQL-like	AI-driven dynamic data sharding utilizing machine learning (Random Forest), supervised learning with workload-based feature inputs, and real-time shard rebalancing.	AI improves sharded database security by monitoring access patterns and detecting abnormalities - AI boosts performance and security. Anomaly prediction reduces data breaches and illegal access, supporting compliance.	15.29% Query Latency Reduction (0.85s to 0.72s). - Throughput up 20% (50,000 to 60,000 TPS). CPU utilization efficiency increased 10% (75.6% to 83.2%). Memory Usage: 12.8% lower (120GB to 104.6GB). IOPS rose 17.34%. Increased load balancing efficiency 18.34%.Shard Rebalancing Time: 46.88% Lower (3.2s to 1.7s).	A cloud-native NoSQL distributed database cluster was used for experiments. The Cloud has 10 nodes with 16 CPUs, 64 GB RAM, and 2 TB storage. Cloud services and e-commerce workloads employ TPC-C and YCSB benchmark datasets.
[34]	Blockchain-based Distributed Ledger Technologies (Ethereum, Hyperledger, Multichain, etc.)	Logic specialized to smart contracts; not a conventional query language	Smart contracts for access regulation, permissioned blockchains for cost efficiency, anchoring mechanisms, hybrid on-chain/off-chain storage, Directed Acyclic Graph (IOTA)	PKI, DID, SSI, smart contract access control, behavior detection, secure firmware upgrades, and cryptographic assurances (immutability, data integrity)	Latency, Throughput, Technical, Operational, Economic Feasibility Cost, security, scalability	Partially employs edge devices and gateways as blockchain nodes, hybrid storage and computation models in cloud.
[35]	Implicitly inferred; presumes AI-driven data storage and processing systems	Not stated; emphasizes AI algorithm processing instead of querying	Quantization, model compression, GPU/TPU, pipelining, parallel processing, distributed AI, edge computing, fog computing	Distributed data integrity, blockchain, decentralized storage, PKI, and data preparation for quality and validity	Latency, energy use, battery life, scalability, real-time responsiveness, AI execution efficiency	For scalability and real-time processing, edge, fog, distributed AI frameworks, and cloud platforms are used extensively.
[36]	Pluggable MQTT framework; bespoke IECC DMF (not DBMS)	Custom data flow, no query language, plugin-based setup	Modular plugin system, low-latency MQTT communication, differential privacy, energy status assessment, centralized configuration, visual management tools, real-time monitoring	TLS for secure MQTT connection, differential privacy plugin, traceability monitoring, adjustable access constraints, centralized security policy enforcement	Battery life, latency, energy use, privacy Tradeoff, Real-Time Validation	Yes; plugin-driven deployment, integrated IoT–Edge–Cloud support with MQTT brokers at each node, verified in real-world cloud and edge scenarios.

[37]	Information-Centric Network (ICN) – Named Data Networking (NDN)	NDN-based interest-based data retrieval rather than a query language	Lifetime classification, selective caching based on node location (edge/betweenness), avoiding short-lived data caching, betweenness centrality, ndnSIM simulation.	Secure content naming and in-network caching are ICN/NDN features not explicitly addressed.	Cache Hit Ratio, Content Retrieval Delay, Data Freshness Effect, Consumer count Scalability	Yes; edge/fog caching nodes, edge-proximity deployment schemes
[38]	Enterprise Data Warehouse (DW) and Data Lake (DL)	SQL for Data Warehousing; schema-on-read and open formats (e.g., Hadoop, Spark) for Deep Learning	DW: ETL, OLAP indexing, pre-aggregation (MOLAP, ROLAP); DL: schema-on-read, ELT, scalable storage, compute-storage separation.	DW: Strong access control, GDPR/HIPAA compliance; DL: Encryption, access control, network security, metadata governance	DW: Query speed, latency, scalability, availability; DL: Ingestion speed, scalability, real-time processing, metadata discoverability.	DW and DL enable cloud deployment, and DL offers configurable on-premise/hybrid/cloud topologies with AWS, Azure, and Google BigLake.
[39]	Access Control System (CP-ABE integrated) on blockchain	Unstandardized query language; attribute-based encryption (CP-ABE) logic.	Bilinear pairing with DBDH for mutual authentication and key agreement, lightweight CP-ABE, efficient protocol architecture, public permissioned blockchain with PBFT.	CP-ABE, secure data upload, session key, mutual authentication, data nonrepudiation, accountability, blockchain auditing, AVISPA/IND-CPA validation, ECC encryption, hash-based ID protection	Computation and Communication Cost (ms& bits), AVISPA security simulation, guessing/tracing/replay/impersonation resilience	Computation and Communication Cost (ms& bits), AVISPA security simulation, guessing/tracing/replay /impersonation resilience
[40]	RFID, sensor streams, metadata storage, heterogeneous IoT data stores	XML/XQuery, semi-structured query languages, and path expressions for hierarchical data	Local indexing, hierarchical nomenclature (IANA, UUID), SOA-based service encapsulation, time-series aggregation, and the use of stream models	Legislation on data protection (e.g. GDPR), encryption protocols, access control mechanisms, policy implementation, privacy classifications	Scalability, Indexing Efficiency, Time-Series Sampling Precision, Transaction Processing Robustness	Indeed; distributed and mixed IoT architectures, edge data proprietorship, private/public network segmentation
[41]	Distributed Data Systems facilitating Digital Twins (including IoT sensors, fog/cloud databases)	Inferred structured/unstructured query usage in AI/BC contexts.	Deep learning, prediction, orchestration, container-based edge orchestration, SDN, Fog Computing, smart data techniques, real-time analytics, lifecycle-based DT design	SDN-based closed-loop security control, traceability, auditability, access policy enforcement, secure data provenance, blockchain-based immutable logs	Fast Response, Data Provenance, Predictive Accuracy, Latency, Bandwidth Optimization, System Scalability	Yes, robust Fog, Edge, and Cloud computing integration enabling Digital Twin synchronization and real-time feedback.
[42]	"Blockchain-based Security Architecture (Hyperledger Fabric with IPFS for extended storage)" n	"Cell for smart contract logic (excluding SQL or conventional queries)"	"Off-chain storage utilizing IPFS, dynamic ECQV certificates, Schnorr zero-knowledge proofs, fog computing-based smart contracts, distributed authentication, and elliptic curve cryptography" cell	"Anonymity, unlinkability, mutual authentication, ECQV certificates, ECIES encryption, Schnorr ZKP, dynamic session key exchange, blockchain audit, resiience to replay/double-spend" cell	"Blockchain Storage Utilization, Search Latency, End-to-End Latency (IoT–Fog–Cloud), Certificate Activation Time, Batch Completion Time" cells	"Yes; integrated with 5G IoT, fog nodes, and cloud CSPs utilizing a decentralized architecture with smart contract execution at edge (fog) nodes."
[43]	Hybrid Architecture (IoT-Fog-Cloud)	Implied endorsement of ML/AI frameworks	Dynamic resource allocation, load balancing, delay reduction, job scheduling	Data anonymization, HMAC/OAuth authentication, AES/RSA encryption	Latency, resource use, decision accuracy, energy efficiency, DSS	Full Cloud and Fog computing capabilities with dynamic layer distribution
[44]	Industrial Cloud/Fog/Edge IoRT Systems	Not expressly specified; incorporates ML/DL and simulation modeling frameworks.	Task scheduling, digital twins, federated learning, swarm cooperation, predictive modeling	Blockchain-based data integrity, semantic access control, failure checking	Fast processing, accurate decisions, data scalability, resilience	Strong cloud, edge, and fog computing support, offloading, and decentralization
[45]	ML-IDMS Intelligent Database Management System	SQL with ML integration	Initializing neural networks, selecting features, reducing dimensionality, decision modeling	ML-enhanced cloud security, anomaly detection, encryption, semantic transformation	Query execution (19.27s), storage efficiency (83.78%), correctness (90%), redundancy reduction (66.42%), throughput (7.93 Gbps), latency (14.4 ms)	Cloud processing with Hadoop HDFS and real-time IoT support

[46]	Dynamic Indexing Framework for Internet of Things Streaming Data	Timestamp-based queries utilizing a bespoke index structure, combined with key-value pairs.	Windowed floating-point compression, bit-padding, Huffman encoding, delta encoding, and zigzag encoding	Not specifically stated; focus on data integrity and deduplication	Compression ratio (2.12%), storage efficiency (97.88%), processing duration (real-time capabilities for <25200 records)	Facilitates streaming and real-time data integration from many IoT sources with Apache Kafka.
[47]	Internet of Things-Driven Health Data Ecosystem (no particular database name)	Not specifically delineated; employs sensor-driven real-time data streams and cloud interfaces.	Data compression, aggregation, and reduction; hybrid processing at the edge, fog, and cloud; machine learning-driven decision support.	Encryption, access control, anomaly detection, and blockchain integration for integrity.	Scalability, latency, decision-making accuracy, resource efficiency, patient outcomes	Facilitates edge computing, fog nodes, and cloud systems by adaptive data offloading.
[48]	Hadoop, Apache Spark are cloud-based distributed systems.	JSON over REST API, Spark SQL, Python	Computing on a cluster in memory, MapReduce, parallel processing with Spark, and RESTful design for node updates	Observations on cloud vulnerabilities and provider-managed security are included, although not in great detail.	Efficiency in batch processing, fault tolerance, horizontal scalability, and speed (real-time stream processing) are all important.	Strong cloud computing focus (SaaS, PaaS, and IaaS); no edge computing characteristics stated.
[49]	IoRT-based Cyber-Physical Systems with Cloud, Fog, and Edge Support	Not specifically mentioned; refers to simulation tools, visual data, digital twins, and machine learning interfaces	Swarm intelligence, federated learning, sensor fusion, real-time monitoring, spatial computing, and digital twin modeling.	Blockchain, contextualized control, semantic technologies, ambient intelligence, and decentralized coordination	Scalability, real-time data accuracy, decision-making speed, routing efficiency, energy efficiency, job execution precision	Fully integrated edge, fog, and cloud computing for collaborative robotics and real-time IoRT data sharing
[50]	ML-IDMS Intelligent Database Managemen t System	SQL with ML integration	Structured Query Language (SQL); extended with ML integration Structured Query Language (SQL); extended ML integration Initializing neural networks, selecting features, reducing dimensionality, decision modeling	ML-enhanced cloud with security, anomaly detection, encryption, semantic transformati on	Query execution (19.27s), storage efficiency (83.78%), correctness (90%), redundancy reduction (66.42%), throughput (7.93 Gbps), latency (14.4 ms).	Cloud processing with Hadoop HDFS and real-time IoT support
[51]	Smart greenhouse management with IoT	Protocols: MQTT, REST, DDS, XMPP, ZigBee; no SQL.	Predictive analytics, LPWAN/LoRaWAN, MAC algorithms, AI/ML decision assistance, fog/cloud/edge synergy, energy load shaping, digital twins	Limited detail on interoperability, structural monitoring, secure IoT protocols, data integrity standards.	Up to 43% energy savings, $500/acre cost savings, data transfer speed, sensor accuracy, greenhouse microclimate management	A cloud-based DSS supports edge computing, cloud and fog integration, low-power WANs, GPS/LEO/5G infrastructure, and structural health monitoring.

This study compares IoT data management database solutions and architectures in a thorough table. It captures technological traits and capabilities that affect a database's IoT-specific performance. Six columns cover database system topics: Database Type & Name, Query Language, Optimization Techniques, Security Mechanisms, Performance Metrics, and Cloud/Edge Support. The first column, “Database Type & Name,” lists the database technology being studied and its name. This encompasses NoSQL, time-series, blockchain, and actor-oriented databases. This categorization of databases shows the range in data processing methodologies, which is necessary to match database capabilities with IoT demands. The second column, “Query Language,” is the main database interface. Others, especially those based on newer paradigms like actor-based models or blockchain, use asynchronous method calls or smart contract logic instead of SQL or MQL or CQL. Understanding the querying interface helps assess a database system's complexity, flexibility, and integration possibilities in IoT applications.

The final column, “Optimization Techniques,” describes database efficiency and scalability strategies. Time-series data compaction, compression, machine learning-based data sharding, and other performance-driven features may be used. Optimization is essential for handling big, high-velocity data in IoT use cases while retaining performance and dependability. The fourth column, “Security Mechanisms,” explains each system's privacy and protection procedures. These include encryption standards (e.g., AES, RSA), authentication methods, identity management systems like CP-ABE, and pseudonyms. Such security techniques protect data during transmission, storage, and access in IoT environments, where data sensitivity and user privacy are important. The fifth column, “Performance Metrics,” lists system performance benchmarks. Latency, throughput, CPU, memory, storage efficiency, and reaction times are measurements. This data lets database systems be compared under different operating circumstances to see whether platforms can fulfill IoT deployments' real-time and resource-constrained needs. The last column, “Cloud/Edge Support,” evaluates each system's cloud and edge computing compatibility. It checks if the database supports distributed deployments, containerized implementations (Kubernetes or Docker), and network edge real-time processing. IoT systems depend on edge computing to reduce latency, bandwidth usage, and data processing responsiveness. These six columns provide a holistic overview of IoT database technologies, helping readers understand how different solutions meet current data-driven environments' technical, performance, and security concerns.

RECOMMENDATIONS

1. **Adopt Hybrid Edge–Fog–Cloud Architectures** Organizations should incorporate hybrid computing architectures to optimize real-time processing and minimize latency. The scalability, backup, and sophisticated analytics capabilities of the cloud and fog layers complement the edge nodes' capacity to process time-sensitive data in real-time. Using this tiered approach improves system responsiveness and resource usage, which is particularly useful for Internet of Things applications that deal with high-frequency data streams.
2. **Leverage AI and Machine Learning for Intelligent Data Handling**
Employ machine learning methods, including anomaly detection, predictive modeling, and dynamic sharding, to improve transactional integrity and automate data management. These models enhance data quality, optimize storage, and provide adaptive responses to evolving data patterns.
3. **Utilize Distributed NoSQL Databases for Scalability** NoSQL systems such as MongoDB, Cassandra, and Redis are more adept than conventional RDBMS at managing diverse and high-volume IoT data. Their schema-less architecture and horizontal scalability meet the performance demands of contemporary IoT ecosystems, particularly when integrated with containerized deployments and orchestration platforms such as Kubernetes.
4. **Implement Privacy-Preserving and Secure Frameworks** To adhere to data protection standards like GDPR, IoT devices must incorporate privacy-preserving identifiers, blockchain-based access restrictions, and encrypted communication protocols. Integrating differential privacy and decentralized identity frameworks will enhance user confidence and data security.
5. **Optimize Time-Series and Streaming Data Handling** Prioritize data indexing, compression, and selective caching for the optimal handling of IoT-generated time-series data. Utilizing models like VTSA (Versioned Time-Series Algebra) and TSBPS (Temporal-Spatial Block Partitioning Strategy) can markedly improve retrieval velocity and storage efficacy.
6. **Standardize Interoperability and Semantic Data Models** To facilitate seamless data transmission and integration among various IoT systems, it is advisable to use standardized data formats, query interfaces, and semantic ontologies. This promotes interoperability and lowers complexity in extensive, diverse IoT implementations.
7. **Promote Research on Lightweight and Energy-Efficient Solutions**
Considering the constrained resources of numerous IoT edge devices, forthcoming advancements should prioritize lightweight algorithms, energy-efficient indexing systems, and model compression methodologies. Research should further investigate adaptive resource management to equilibrate performance and energy usage.
8. **Enhance Integration with Business Intelligence Tools** Predictive, automated, and real-time decision-making is made possible by bridging Internet of Things platforms with powerful business intelligence systems. Business intelligence frameworks that use artificial intelligence and big data analytics can assist firms in gaining deeper insights and strategic benefits.

CONCLUSION

The findings of this research have shown that in order to effectively handle data from the Internet of Things (IoT), database systems that are capable of exceeding the capabilities of standard relational models are required. It is clear that current solutions are increasingly suited for the real-time, scalable, and diverse nature of Internet of Things settings. This is visible via a systematic comparison of advanced database designs, which includes NoSQL, time-series, actor-oriented, and block chain-based systems. Under high-concurrency situations, performance may be considerably improved by utilizing optimization techniques such as sharding that is driven by machine learning, indexing that is based on streaming, and compression algorithms. Encryption, attribute-based access control, and blockchain auditing are some of the sophisticated security measures that may be integrated into the Internet of Things (IoT) to ensure the protection and integrity of critical data. When it comes to lowering latency and allowing localized data processing, the capability of these systems to function across cloud, fog, and edge infrastructures is very necessary. All things considered, the comparative insights that are presented in this article serve a helpful reference for selecting and creating database systems that are in accordance with the particular requirements of Internet of Things applications. Future research should investigate lightweight, energy-efficient database solutions that seamlessly incorporate artificial intelligence, edge computing, and privacy-preserving strategies in order to satisfy the expanding demands of smart, connected settings. This is because infrastructures for the Internet of Things (IoT) continue to develop in both complexity and size.

REFERENCES

[1] A. Pandya, C. Kulkarni, K. Mali, and J. Wang, "An Open Source Cloud-Based NoSQL and NewSQL Database Benchmarking Platform for IoT Data," in Benchmarking, Measuring, and Optimizing: First BenchCouncil International Symposium, Bench 2018, Seattle, WA, USA, December 10-13, 2018, Revised Selected Papers 1, Springer, 2019, pp. 65–77.

[2] K. Mahmood, T. Risch, and K. Orsborn, "Analytics of iiot data using a nosql datastore," in 2021 IEEE International Conference on Smart Computing (SMARTCOMP), IEEE, 2021, pp. 97–104.

[3] V. Pupezescu, M.-C. Pupezescu, and L.-A. Perișoară, "Optimizations of Database Management Systems for Real Time IoT Edge Applications," in 2022 23rd International Carpathian Control Conference (ICCC), IEEE, 2022, pp.171–176.

[4] K. Dubey, A. Raj, A. K. Singh, and S. K. Tripathi, "An Efficient Data Management Policy For IoT Application with Tangential Data Control System in a Cloud Fog Environment," in 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, 2024, pp. 1–6.

[5] A. M. Al-madni, X. Ying, M. Tawfik, and Z. A. T. Ahmed, "An Optimized Blockchain Model for Secure and Efficient Data Management in Internet of Things," in 2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS), IEEE, 2024, pp. 1–11.

[6] S. G. Thandekkattu, G. T. Ayem, and S. Rakshit, "Data Management and Analytics in the Internet of Things (IoT)," in Designing Sustainable Internet of Things Solutions for Smart Industries, IGI Global, 2025, pp. 209–228.

[7] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Performance of databases in IoT applications," in 2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), IEEE, 2020, pp. 190–192.

[8] M. M. Nashipudmath, V. Chitre, S. Shinde, and G. Phade, "Smart Data Management in IoT: Leveraging Wireless Sensor Networks for Efficient Information Processing.," Journal of Electrical Systems, vol. 19, no. 2, 2023.

[9] T. K. Adenekan, "Revolutionizing Business Intelligence: Integrating Big Data, AI, Machine Learning, IoT, and Blockchain for Smarter Decision-Making." [Online]. Available: https://www.researchgate.net/publication/389340232

[10] Q. T. Doan, A. S. M. Kayes, W. Rahayu, and K. Nguyen, "Integration of IoT Streaming Data with Efficient Indexing and Storage Optimization," IEEE Access, vol. 8, pp. 47456–47467, 2020, doi: 10.1109/ACCESS.2020.2980006.

[11] N. Shirvanian, M. Shams, and A. M. Rahmani, "Internet of Things data management: A systematic literature review, vision, and future trends," International Journal of Communication Systems, vol. 35, no. 14, p. e5267, 2022.

[12] L. Andrade, C. J. L. De Santana, B. D. M. Alencar, C. J. Silva, and C. Prazeres, "Data interplay: A model to optimize data usage in the Internet of Things," SoftwPract Exp, vol. 53, no. 6, pp. 1410–1437, 2023.

[13] S. Tamilselvan, M. Babu, P. Rajasekar, K. Kalaiselvi, R. Shanmugam, and V. Kadumbadi, "Enhancing IoT Network Capabilities through Edge Computing Strategies for Real-Time Data Analytics and Efficient Information Processing," in 2024 10th International Conference on Communication and Signal Processing (ICCSP), IEEE, 2024, pp. 306–310.

[14] V. Pupezescu, M.-C. Pupezescu, and L.-A. Perișoară, "Optimizations of Database Management Systems for Real Time IoT Edge Applications," in 2022 23rd International Carpathian Control Conference (ICCC), IEEE, 2022, pp. 171–176.

[15] Q.-T. Doan, A. S. M. Kayes, W. Rahayu, and K. Nguyen, "A framework for iot streaming data indexing and query optimization," IEEE Sens J, vol. 22, no. 14, pp. 14436–14447, 2022.

[16] A. K. Dinkar, M. A. Haque, and A. K. Choudhary, "Enhancing IoT Data Analysis with Machine Learning: A Comprehensive Overview," LatIA, vol. 2, p. 9, 2024.

[17] V. Juneja et al., "Optimization-Based Data Science for an IoT Service Applicable in Smart Cities," in Handbook of Research on Data-Driven Mathematical Modeling in Smart Cities, IGI Global, 2023, pp. 300–321.

[18] M. Sathesh, K. Ramakrishnan, M. Raja, K. Kalaiarasi, and M. Balamurugan, "Edge Computing Integration in IoT Networks for Real-Time Data Processing," in 2024 International Conference on Cybernation and Computation (CYBERCOM), IEEE, 2024, pp. 585–590.

[19] Mauli Fathia Zahra and Muhammad IrwanPadli Nasution, "Manajemen Data Real-Time UntukAplikasi Internet Of Things (IOT)," JURNAL PENELITIAN SISTEM INFORMASI (JPSI), vol. 2, no. 2, pp. 111–120, May 2024, doi: 10.54066/jpsi.v2i2.1917.

[20] O. Farooq, P. Singh, M. Hedabou, W. Boulila, and B. Benjdira, "Machine Learning Analytic-Based Two-Staged Data Management Framework for Internet of Things," Sensors, vol. 23, no. 5, Mar. 2023, doi: 10.3390/s23052427.

[21] M. Akil, L. Islami, S. Fischer-Hübner, L. A. Martucci, and A. Zuccato, "Privacy-preserving identifiers for IoT: A systematic literature review," 2020, Institute of Electrical and Electronics Engineers Inc.doi: 10.1109/ACCESS.2020.3023659.

[22] H. Gadde, "Optimizing Transactional Integrity with AI in Distributed Database Systems," 2024.

[23] T. K. Adenekan, "Revolutionizing Business Intelligence: Integrating Big Data, AI, Machine Learning, IoT, and Blockchain for Smarter Decision-Making." [Online]. Available: https://www.researchgate.net/publication/389340232

[24] M. M. Eyada, W. Saber, M. M. El Genidy, and F. Amer, "Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments," IEEE Access, vol. 8, pp. 110656–110668, 2020, doi: 10.1109/ACCESS.2020.3002164.

[25] "ssrn-3399649".

[26] T. Gkamas, V. Karaiskos, and S. Kontogiannis, "Performance Evaluation of Distributed Database Strategies Using Docker as a Service for Industrial IoT Data: Application to Industry 4.0," Information (Switzerland), vol. 13, no. 4, Apr. 2022, doi: 10.3390/info13040190.

[27] J. Du, "Real-Time Information Exchange Strategy for Large Data Volumes Based on IoT," ComputIntellNeurosci, vol. 2022, 2022, doi: 10.1155/2022/2882643.

[28] M. Akil, L. Islami, S. Fischer-Hübner, L. A. Martucci, and A. Zuccato, "Privacy-preserving identifiers for IoT: A systematic literature review," 2020, Institute of Electrical and Electronics Engineers Inc.doi: 10.1109/ACCESS.2020.3023659.

[29] S. Zhou, J. Sun, K. Xu, and G. Wang, "AI-Driven Data Processing and Decision Optimization in IoT through Edge Computing and Cloud Architecture," Journal of AI-Powered Medical Innovations (International online ISSN 3078-1930), vol. 2, no. 1, pp. 64–92, Dec. 2024, doi: 10.60087/vol2iisue1.p006.

[30] L. B. Dias, M. Holanda, R. C. Huacarpuma, and R. T. De Sousa, "NoSQL database performance tuning for IoT data cassandra case study," in IoTBDS 2018 - Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security, SciTePress, 2018, pp. 277–284. doi: 10.5220/0006782702770284.

[31] R. Kang and S. Song, "Optimizing Time Series Queries with Versions," Proceedings of the ACM on Management of Data, vol. 2, no. 3, pp. 1–27, May 2024, doi: 10.1145/3654962.

[32] Y. Wang, J. C. Dos Reis, K. M. Borggren, M. A. Vaz Salles, C. B. Medeiros, and Y. Zhou, "Modeling and building IoT data platforms with actor-oriented databases," in Advances in Database Technology - EDBT, OpenProceedings.org, 2019, pp. 512–523. doi: 10.5441/002/edbt.2019.47.

[33] H. Gadde, "AI in Dynamic Data Sharding for Optimized Performance in Large Databases," 2022.

[34] S. K. Lo et al., "Analysis of Blockchain Solutions for IoT: A Systematic Literature Review," IEEE Access, vol. 7, pp. 58822–58835, 2019, doi: 10.1109/ACCESS.2019.2914675.

[35] P. Yuan, "Artificial intelligence in the Internet of Things: Integrating and optimizing AI algorithms for real-time data processing and decision-making," Applied and Computational Engineering, vol. 102, no. 1, pp. 84–89, Nov. 2024, doi: 10.54254/2755-2721/102/20241015.

[36] J. Judvaitis, E. Blumbergs, A. Arzovs, A. I. Mackus, R. Balass, and L. Selavo, "A Set of Tools and Data Management Framework for the IoT–Edge–Cloud Continuum," Applied System Innovation, vol. 7, no. 6, Dec. 2024, doi: 10.3390/asi7060130.

[37] M. S. Zahedinia, M. R. Khayyambashi, and A. Bohlooli, "IoT data management for caching performance improvement in NDN," Sep. 01, 2023. doi: 10.21203/rs.3.rs-3305411/v1.

[38] A. Nambiar and D. Mundra, "An Overview of Data Warehouse and Data Lake in Modern Enterprise Data Management," Dec. 01, 2022, MDPI. doi: 10.3390/bdcc6040132.

[39] J. Y. Lee et al., "Blockchain-Based Data Access Control and Key Agreement System in IoT Environment," Sensors, vol. 23, no. 11, Jun. 2023, doi: 10.3390/s23115173.

[40] J. Cooper and A. James, "Challenges for database management in the internet of things," IETE Technical Review (Institution of Electronics and Telecommunication Engineers, India), vol. 26, no. 5, pp. 320–329, Sep. 2009, doi: 10.4103/0256-4602.55275.

[41] G. Poornima, V. Janardhanachari, and D. S. Sakkari, "Data Management for IoT and Digital Twin," 2022, pp. 28–45. doi: 10.4018/978-1-6684-5722-1.ch002.

[42] T. Hewa, A. Braeken, M. Liyanage, and M. Ylianttila, "Fog Computing and Blockchain-Based Security Service Architecture for 5G Industrial IoT-Enabled Cloud Manufacturing," IEEE Trans Industr Inform, vol. 18, no. 10, pp. 7174–7185, Oct. 2022, doi: 10.1109/TII.2022.3140792.

[43] A. A. Al-Atawi, "Enhancing data management and real-time decision making with IoT, cloud, and fog computing," IET Wireless Sensor Systems, Dec. 2024, doi: 10.1049/wss2.12099.

[44] M. Andronieet al., "Big Data Management Algorithms, Deep Learning-Based Object Detection Technologies, and Geospatial Simulation and Sensor Fusion Tools in the Internet of Robotic Things," Feb. 01, 2023, MDPI. doi: 10.3390/ijgi12020035.

[45] R. Clavijo-López, W. A. Luy Navarrete, J. M. Velásquez, C. M. Aguilar Saldaña, A. M. Ocas, and C. A. Flores Tananta, "Integrating Novel Machine Learning for Big Data Analytics and IoT Technology in Intelligent Database Management Systems," Journal of Internet Services and Information Security, vol. 14, no. 1, pp. 206–218, Feb. 2024, doi: 10.58346/JISIS.2024.I1.014.

[46] Q. T. Doan, A. S. M. Kayes, W. Rahayu, and K. Nguyen, "Integration of IoT Streaming Data with Efficient Indexing and Storage Optimization," IEEE Access, vol. 8, pp. 47456–47467, 2020, doi: 10.1109/ACCESS.2020.2980006.

[47] M. Z. Hussain et al., "IoT Data Management and A Brief Analysis of IoT in the Health Industry," in 2023 Computer Applications and Technological Solutions, CATS 2023, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/CATS58046.2023.10424074.

[48] S. Gupta and R. Godavarti, "IoT data management using cloud computing and big data technologies," International Journal of Software Innovation, vol. 8, no. 4, pp. 50–58, Oct. 2020, doi: 10.4018/IJSI.2020100104.

[49] D. Gamero, A. Dugenske, C. Saldana, T. Kurfess, and K. Fu, "Scalability Testing Approach for Internet of Things for Manufacturing SQL and NoSQL Database Latency and Throughput," J Comput Inf Sci Eng, vol. 22, no. 6, Dec. 2022, doi: 10.1115/1.4055733.

[50] X. Ma, "The analysis of the internet of things database query and optimization using deep learning network model," PLoS One, vol. 19, no. 6 June, Jun. 2024, doi: 10.1371/journal.pone.0306291.

[51] C. Maraveas, D. Piromalis, K. G. Arvanitis, T. Bartzanas, and D. Loukatos, "Applications of IoT for optimized greenhouse environment and resources management," Jul. 01, 2022, Elsevier B.V.doi: 10.1016/j.compag.2022.106993.

****